

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Tuan Huu Pham et al. Art Unit : 2126  
Serial No. : 09/855,683 Examiner : The T. Ho  
Filed : May 16, 2001  
Title : COMPONENT INSTALLATION TOOL

**Mail Stop Appeal Brief - Patents**  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

BRIEF ON APPEAL

TABLE OF CONTENTS

<u>Section</u>	<u>Page(s)</u>
I. Real Party in Interest.....	2
II. Related Appeals and Interferences.....	3
III. Status of Claims.....	4
IV. Status of Amendments.....	5
V. Summary of Claimed Subject Matter .....	6-10
VI. Grounds of Rejection.....	11
VII. Argument.....	12-16
VIII. Claims Appendix.....	17-39
IX. Evidence Appendix .....	40
X. Related Proceedings Appendix .....	41

03/24/2005 JADD01 00000004 09855683

01 FC:1402 500.00 OP

Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 2 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

**I. Real Party in Interest**

America Online, Inc., the assignee of this application, is the real party is interest.

Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 3 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

## **II. Related Appeals and Interferences**

There are no related appeals or interferences.

### **III. Status of Claims**

Claims 1, 8-14, 17, 18, 23-27, 30-32, 39-42, 45, 46, 51-55, 58-60, 67-70, 73, 74, 79-83, and 86-139 are pending and stand rejected, with claims 1, 32, 60, 88, 105, and 120 being independent.

Claims 2-7, 15, 16, 19-22, 28, 29, 33-38, 43, 44, 47-50, 56, 57, 61-66, 71, 72, 75-78, 84, and 85 are cancelled.

Appellants have appealed the rejection of claims 8-14, 17, 18, 23-27, 30-32, 39-42, 45, 46, 51-55, 58-60, 67-70, 73, 74, 79-83, and 86-139.

Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 5 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

#### **IV. Status of Amendments**

Claims 126 and 128 were amended after final to change their respective dependencies.  
The Advisory Action mailed January 18, 2005 stated that the amendments would be entered for the purposes of appeal.

## **V. Summary of Claimed Subject Matter**

The following summarizes disclosure related to each independent claim with references to the application specification and drawings. The references to the specification and drawings are meant to be exemplary, and not limiting.

### **Independent claim 1**

Independent claim 1 is directed to a system for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices. As shown in Figs. 6a and 6b, a component installation tool 622 includes a receiving module 6221, a detection module 6222, and an installation module 6223, to receive, detect and install connectivity components on a client device 620 to enable connectivity to a host system 610. See specification, p. 17, line 12 to p. 18, line 13.

The receiving module 6221 is structured and arranged to receive a connectivity component that enables connectivity to a host system 610 by at least one of several different hardware devices. See specification, p. 18, lines 15-18. The detection module 6222 is structured and arranged to detect whether installation of the connectivity component is needed to enable connectivity between the client device 620 and the host system 610 using a selected hardware device. See specification, p. 18, lines 18-21. The installation module 6223 is structured and arranged to install the connectivity component when the connectivity component is needed to enable connectivity between the client device 620 and the host system 610 using the selected hardware device. See specification, p. 18, lines 22-25.

More specifically, the detection module 6222 is structured and arranged to detect a new hardware device and, based on detecting the new hardware device, to determine whether a connectivity component is stored locally that is needed to enable connectivity between the client device 620 and the host system 610 using the new hardware device. See specification, p. 20, lines 26-29. The receiving module 6221 is structured and arranged to receive an updated connectivity component from a remote server when the detection module 6222 determines that the connectivity component that is needed to enable connectivity between the client device 620 and the host system 610 using the new hardware device is not stored locally and the installation

module 6223 installs the updated connectivity component. See specification, p. 20, line 31 to p. 21, line 2.

### **Independent claim 32**

Independent claim 32 is directed to a method for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices. As shown in Fig. 7, the method includes receiving a connectivity component that enables connectivity to a host system by at least one of several different hardware devices (step 710), detecting whether installation of the connectivity component is needed to enable connectivity between the client device and the host system using a selected hardware device (step 720), and installing the connectivity component when connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device (step 730). See specification, p. 22, lines 18-26.

The method also includes detecting a new hardware device and, based on detecting the new hardware device, determining whether a connectivity component is stored locally that is needed to enable connectivity between the client device and the host system using the new hardware device. When an updated connectivity component that is needed to enable connectivity is not stored locally, an updated connectivity component is received from a remote server and the updated connectivity component received from the remote server is installed. See specification, p. 25, lines 5-13.

### **Independent claim 60**

Independent claim 60 is directed to a computer program for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, where the computer program is stored on a computer readable medium or is a propagated signal. A component installation tool 622 includes one or more software components capable of functioning in the manner described below. The specification and figures refer to modules of the component installation tool, which may be implemented as software components. See specification, p. 4, lines 19-20, p. 17, lines 12-15, and p. 28, line 26 to p. 29, line 17. As shown in Figs. 6a and 6b, a component installation tool

622 includes a receiving module 6221, a detection module 6222, and an installation module 6223, to receive, detect and install connectivity components on a client device 620 to enable connectivity to a host system 610. See specification, p. 17, line 12 to p. 18, line 13.

A receiving code segment (receiving module 6221) causes the computer to receive a connectivity component that enables connectivity to a host system 610 by at least one of several different hardware devices. See specification, p. 18, lines 15-18. A detection code segment (detection module 6222) causes the computer to detect whether installation of the connectivity component is needed to enable connectivity between the client device 620 and the host system 610 using a selected hardware device. See specification, p. 18, lines 18-21. An installation code segment (installation module 6223) causes the computer to install the connectivity component when the connectivity component is needed to enable connectivity between the client device 620 and the host system 610 using the selected hardware device. See specification, p. 18, lines 22-25.

The detection code segment (detection module 6222) causes the computer to detect a new hardware device and, based on detecting the new hardware device, to determine whether a connectivity component is stored locally that is needed to enable connectivity between the client device 620 and the host system 610 using the new hardware device. See specification, p. 20, lines 26-29. The receiving code segment (receiving module 6221) causes the computer to receive an updated connectivity component from a remote server when the detection code segment (detection module 6222) determines that the connectivity component that is needed to enable connectivity between the client device 620 and the host system 610 using the new hardware device is not stored locally and the installation code segment (installation module 6223) installs the updated connectivity component. See specification, p. 20, line 31 to p. 21, line 2.

### **Independent claim 88**

Independent claim 88 is directed to a system for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices. As shown in Figs. 6a and 6b, a component installation tool 622 includes a receiving module 6221, a detection module 6222, and an installation module 6223, to receive,



detect and install connectivity components on a client device 620 to enable connectivity to a host system 610. See specification, p. 17, line 12 to p. 18, line 13.

The receiving module 6221 is structured and arranged to receive multiple connectivity components that enable connectivity to a host system 610 by at least one of several different hardware devices, where the receiving module 6221 is structured and arranged to copy the connectivity components to the client device 620 from a compact disk and store the connectivity components in a dormant state on the client device 620. See specification, p. 18, lines 26-30 and p. 19, lines 4-11.

The detection module 6222 is structured and arranged to detect whether installation of at least one of the connectivity components is needed to enable connectivity between the client device and the host system using a selected hardware device. See specification, p. 19, lines 14-25. The installation module 6223 is structured and arranged to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device. See specification, p. 18, lines 21-25.

#### **Independent claim 105**

Independent claim 105 is directed to a method for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices. As shown in Fig. 7, the method includes receiving multiple connectivity components that enable connectivity to a host system by at least one of several different hardware devices, where receiving the connectivity components includes copying the connectivity components to the client device from a compact disk and storing the connectivity components in a dormant state on the client device (step 710). See specification, p. 23, lines 15-25. The method includes detecting whether installation of at least one of the connectivity components is needed to enable connectivity between the client device and the host system using a selected hardware device (step 720) and installing the connectivity component when connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device (step 730). See specification, p. 22, lines 18-26 and p. 23, line 15 to p. 24, line 6.

### **Independent claim 120**

Independent claim 120 is directed to a computer program for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, where the computer program is stored on a computer readable medium or is a propagated signal. A component installation tool 622 includes one or more software components capable of functioning in the manner described below. The specification and figures refer to modules of the component installation tool, which may be implemented as software components. See specification, p. 4, lines 19-20, p. 17, lines 12-15, and p. 28, line 26 to p. 29, line 17. As shown in Figs. 6a and 6b, a component installation tool 622 includes a receiving module 6221, a detection module 6222, and an installation module 6223, to receive, detect and install connectivity components on a client device 620 to enable connectivity to a host system 610. See specification, p. 17, line 12 to p. 18, line 13.

A receiving code segment (receiving module 6221) causes the computer to receive multiple connectivity components that enable connectivity to a host system 610 by at least one of several different hardware devices, where the receiving code segment (receiving module 6221) causes the computer to copy the connectivity components to the client device 620 from a compact disk and store the connectivity components in a dormant state on the client device 620. See specification, p. 18, lines 26-30 and p. 19, lines 4-11.

A detection code segment (detection module 6222) causes the computer to detect whether installation of at least one of the connectivity components is needed to enable connectivity between the client device and the host system using a selected hardware device. See specification, p. 19, lines 14-25. An installation code segment (installation module 6223) causes the computer to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device. See specification, p. 18, lines 21-25.

## **VI. Grounds of Rejection**

A. Claims 126 and 128 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite.

B. Claims 1, 8-14, 23-27, 30-32, 39-42, 51-55, 58-60, 67-70, 79-83, 86-94, 97-111, 114-126, and 129-139 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Krishnan (U.S. Patent No. 6,075,863) in view of Onosaka (U.S. Patent No. 5,961,608).

C. Dependent claims 17, 18, 45, 46, 73, 74, 95, 96, 112, 113, 127, and 128 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Krishnan in view of Onosaka and further in view of Coutts (U.S. Patent No. 6,311,165).

## **VII. Argument**

### **A. Claims 126 and 128 are not indefinite under 35 U.S.C. § 112, second paragraph.**

Appellants have amended claim 126 to depend from claim 125 and have amended claim 128 to depend from claim 127, as suggested by the Examiner on page 2 in the Final Office Action mailed August 24, 2004. As such, Appellants respectfully request reversal of the rejection of claims 126 and 128.

### **B. Claims 1, 8-14, 23-27, 30-32, 39-42, 51-55, 58-60, 67-70, 79-83, 86-94, 97-111, 114-126, and 129-139 stand would not have been obvious over Krishnan in view of Onosaka.**

#### **1). Claims 1, 8-14, 23-27, 30-32, 39-42, 51-55, 58-60, 67-70, 79-83, 86, and 87**

Appellants request reversal of the rejection because Krishnan and Onosaka, either alone or in combination, fail to describe or suggest these features. Specifically, neither Krishnan nor Onosaka describes or suggests detecting a new hardware device and, based on detecting the new hardware device, determining whether an updated connectivity component is stored locally, as recited in claims 1, 32, and 60. Nor do either of Krishnan or Onosaka describe or suggest receiving an updated connectivity component from a remote server when the updated connectivity component is not stored locally, as recited in claims 1, 32, and 60.

Instead, as relied upon on page 14 of the Final Office Action, Krishnan describes using a data encryption applet between two modems. If the modems contain the same applet version, then a data transfer occurs. If the modems do not contain the same applet version (i.e., one modem has a relatively outdated applet version), then the modems may negotiate to transfer the new applet version among the modems.

"Loading an applet from a remote modem provides a facility for automatic data encryption services. For example, any time a connection is established between a modem 10 and a similar remote modem, the use of a data encryption applet may be negotiated. If the modems already contain the same version of the encryption applet then data transfers may begin. If, however, one of the modems lacks the encryption applet, or has an older, out-of-date version, the modems may negotiate to transfer the new version." See Krishnan, col. 4, line 62 to col. 5, line 3.

As such, Krishnan is premised on the existence and prior installation of modems and their corresponding data encryption applets, which are either shared or updated. Nowhere does Krishnan describe or suggest receiving an updated connectivity component from a remote server, when required connectivity is not detected on the client device at which a new hardware device has been detected. In Krishnan, the transfer of a new applet version is triggered by one of the modems having an old version, and not by the detection of a new hardware device.

In the Final Office Action mailed August 24, 2004, the Examiner acknowledges this shortcoming of Krishnan, and thus turns to Onosaka. See page 5, lines 13-14. Even assuming for the sake of argument that the teachings of Krishnan and Onosaka are properly combinable under 35 U.S.C. 103, their proposed combination still does not describe or suggest the recited nexus of inspecting for and supplementing a client device with a missing updated connectivity component from a remote server based on the detection of the new hardware device.

The Office Action accurately points out that Onosaka describes a process for detecting new hardware devices. However, the Office Action fails to recognize that Onosaka does not describe or suggest receiving an updated connectivity component from a remote server when a connectivity component needed for newly detected hardware is not found on the client. Rather, Onosaka merely changes a pointer in the client to point to the currently selected modem driver.

"Based on the user's actions, the system software dynamically changes the pointers to the currently selected modem driver and/or the currently selected serial driver." See Onosaka, col. 4, lines 34-37.

Moreover, the Response to Arguments on page 14 of the Final Office Action makes it clear that the recited claim is not being considered as a whole, because only a portion of the Applicants' arguments has been considered. The Final Office Action mischaracterizes the recited claims and Applicants' arguments when it states:

"Applicant argued that the cited references do not teach downloading the connectivity component from the remote computer after detecting the component is not in the client (Remarks, first completed paragraph page 28)." See Final Office Action mailed August 24, 2004, page 14.

The Office Action's characterization of the claim and Applicants' arguments makes no mention that these events are triggered by the detection of a new hardware device. Here is what the first complete paragraph from page 28 of the Reply to the Office Action actually stated:

"Thus, Krishnan and Onosaka do not describe or suggest downloading a connectivity component from a remote computer in response to the automatic detection of a new hardware device, after determining that the connectivity component is not present on the client device, as recited in amended claims 1, 32, and 60." See Amendment in Reply to Action of January 16, 2004, filed May 17, 2004, page 28 (emphasis in original).

It is clear from the misstatement of the Applicants' arguments in the Final Office Action that all of the features of the recited claims are not being considered.

Furthermore, there is no motivation or suggestion to modify Krishnan with the teachings of Onosaka. The Final Office Action reasons that the convenience of using a suitable modem from multiple modems to connect to a remote computer, as taught by Onosaka, provides the motivation to modify Krishnan.

"It would have been obvious to apply the teachings of Onosaka to the system of Krishnan because this gives the client computer the convenience of using a suitable modem from multiple modems to connect to a remote computer as disclosed by Onosaka (lines 29-37 column 4)." See Final Office Action mailed August 24, 2004, pages 5-6.

However, this alleged motivation to modify Krishnan with Onosaka for reasons of convenience does not provide the required motivation or suggestion to modify Krishnan with the teachings of Onosaka. Since Krishnan is premised on the existence and prior installation of modems and their corresponding data encryption applets, the convenience of using a suitable modem from multiple modems is not relevant to Krishnan. Krishnan already has an existing modem to connect to a remote computer. Thus, there is no motivation or suggestion to modify Krishnan with Onosaka.

For at least these reasons, Appellants respectfully request reversal of the rejection of independent claims 1, 32, and 60, and their respective dependent claims.

**2). Claims 88-94, 97-111, 114-126, and 129-139**

Appellants request reversal of these claims because Krishnan and Onosaka, either alone or in combination, fail to describe or suggest copying multiple connectivity components from a

compact disk and installing at least one of the connectivity components on the client device when it is needed to enable connectivity.

The Final Office Action relies upon Krishnan at col. 2, lines 56-57 to suggest receiving connectivity components from a disk. See, Final Office Action mailed August 24, 2004, page 15. However, this argument and reasoning are flawed for at least two reasons. First, the relied upon section of Krishnan describes ROM 22, which is a component of modem 10. ROM 22 is not a compact disk, but rather read only memory that is a fixed component of the modem 10. Second, ROM 22 does not store or provide any connectivity components to the client device. Rather, ROM 22 "includes programming for controlling overall operation of modem 10, and for executing applets stored in RAM 20." See Krishnan, col. 3, lines 3-6. Thus, it is clear that Krishnan does not describe or suggest receiving and copying multiple connectivity components from a compact disk, as recited in independent claims 88, 105, and 120.

Onosaka does not remedy this failure of Krishnan and, notably, is not relied upon by the Final Office Action to describe or suggest these features.

Furthermore, there is no motivation or suggestion to modify Krishnan with the teachings of Onosaka, as discussed above.

For at least these reasons, Appellants respectfully request reversal of the rejection of independent claims 88, 105, and 120, and their respective dependent claims.

C. Dependent claims 17, 18, 45, 46, 73, 74, 95, 96, 112, 113, 127, and 128 would not have been obvious over Krishnan in view of Onosaka and further in view of Coutts.

With respect to dependent claims 17, 18, 45, 46, 73, 74, 95, 96, 112, 113, 127, and 128, Appellants respectfully request reversal of the rejection because Krishnan, Onosaka, and Coutts, either alone or in combination, fail to describe or suggest the features discussed above with respect to the independent claims upon which these claims depend. Coutts does not remedy the failure of Krishnan and Onosaka to describe the features discussed above with respect to the independent claims and, notably, is not relied upon in the Final Office Action to support these features.

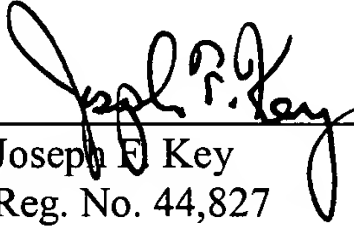
Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 16 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

A check in the amount of \$620 is enclosed of which \$500 is for the brief fee and \$120 is for the extension of time fee. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: 3/23/2005

  
\_\_\_\_\_  
Joseph F. Key  
Reg. No. 44,827

Fish & Richardson P.C.  
1425 K Street, N.W.  
11th Floor  
Washington, DC 20005-3500  
Telephone: (202) 783-5070  
Facsimile: (202) 783-2331



### **VIII. Claims Appendix**

1. (previously presented) A system for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, comprising:

a receiving module that is structured and arranged to receive a connectivity component that enables connectivity to a host system by at least one of several different hardware devices;

a detection module that is structured and arranged to detect whether installation of the connectivity component is needed to enable connectivity between the client device and the host system using a selected hardware device; and

an installation module that is structured and arranged to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device, wherein:

the detection module is structured and arranged to detect a new hardware device and, based on detecting the new hardware device, to determine whether a connectivity component is stored locally that is needed to enable connectivity between the client device and the host system using the new hardware device;

the receiving module is structured and arranged to receive an updated connectivity component from a remote server when the detection module determines that the connectivity component that is needed to enable connectivity between the client device and the host system using the new hardware device is not stored locally; and

the installation module is structured and arranged to install the updated connectivity component received from the remote server.

2-7. (canceled).

8. (previously presented) The system of claim 1 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the new hardware device associated with the device driver.

9. (previously presented) The system of claim 1 wherein the connectivity component is capable of interfacing directly with the new hardware device to enable connectivity between the client device and the host system using the new hardware device.

10. (previously presented) The system of claim 1 wherein the connectivity component includes computer software that interfaces with a driver for the new hardware device that is used to connect to the host system.

11. (original) The system of claim 1 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

12. (original) The system of claim 11 wherein the connectivity component includes a DSL connectivity component to enable connectivity to the host system using a DSL modem.

13. (original) The system of claim 11 wherein the connectivity component includes a cable connectivity component to enable connectivity to the host system using a cable modem.

14. (original) The system of claim 11 wherein the connectivity component includes a satellite connectivity component to enable connectivity to the host system using a satellite modem.

15. (canceled).

16. (canceled).

17. (original) The system of claim 1 wherein the installation module installs a list of programs needed to install the connectivity component.

18. (original) The system of claim 17 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

19-22. (canceled).

23. (previously presented) The system of claim 1 wherein the receiving module, the detection module, and the installation module are structured and arranged to perform automatically without user intervention to receive, detect, and install the connectivity component to enable connectivity to the host system using the new hardware device.

24. (original) The system of claim 1 wherein the connectivity component received includes an updated version of a connectivity component stored on the client device before the connectivity component is installed by the installation module.

25. (original) The system of claim 24 wherein the detection module is structured and arranged to detect whether installation of the connectivity component is needed on the client device by comparing a version of the updated connectivity component received with a version of the connectivity component stored on the client device.

26. (original) The system of claim 1 wherein:  
the receiving module is structured and arranged to receive version information from a remote server associated with a connectivity component;  
the detection module is structured and arranged to detect whether installation of the connectivity component is needed on the client device, and to determine a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received by the receiving module when installation of the connectivity component is needed; and  
the installation module is structured and arranged to install the connectivity component stored on the client device when the detection module determines the version information

associated with the stored connectivity component is correct when compared against the version information received from the remote server.

27. (original) The system of claim 26 wherein:

the receiving module is structured and arranged to receive an updated connectivity component from the remote server when the detection module determines that the version information associated with the connectivity component stored on the client device is not correct when compared against the version information received from the remote server; and

the installation module is structured and arranged to install the updated connectivity component received from the remote server.

28. (canceled).

29. (canceled).

30. (original) The system of claim 1 wherein:

the receiving module is structured and arranged to include a host system receiving module that is structured and arranged to receive a request to send a connectivity component to a local client device; and

the installation module is structured and arranged to include a host system installation module that is structured and arranged to send the connectivity component to the local client device for installation on the local client device in response to the request.

31. (original) The system of claim 30 wherein the detection module is structured and arranged to include a host system detection module that is structured and arranged to determine a version of the connectivity component needed for installation on the local client device.

32. (previously presented) A method for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, the method comprising:

receiving a connectivity component that enables connectivity to a host system by at least one of several different hardware devices;

detecting whether installation of the connectivity component is needed to enable connectivity between the client device and the host system using a selected hardware device; and

installing the connectivity component when connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device, wherein:

detecting whether installation of the connectivity component is necessary includes detecting a new hardware device and, based on detecting the new hardware device, determining whether a connectivity component is stored locally that is needed to enable connectivity between the client device and the host system using the new hardware device;

receiving the connectivity component includes receiving an updated connectivity component from a remote server when the connectivity component that is needed to enable connectivity is not stored locally; and

installing the connectivity component includes installing the updated connectivity component received from the remote server.

33-38. (canceled).

39. (previously presented) The method as in claim 32 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the new hardware device associated with the device driver.

40. (previously presented) The method as in claim 32 wherein the connectivity component is capable of interfacing directly with the new hardware device to enable connectivity between the client device and the host system using the new hardware device.

41. (previously presented) The method as in claim 32 wherein the connectivity component includes computer software that interfaces with a driver for the new hardware device that is used to connect to the host system.

42. (original) The method as in claim 32 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

43. (canceled).

44. (canceled).

45. (original) The method as in claim 32 wherein installing the connectivity component includes installing a list of programs needed to install the connectivity component.

46. (original) The method as in claim 45 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

47-50. (canceled).

51. (previously presented) The method as in claim 32 wherein receiving the connectivity component, detecting whether installation of the connectivity component is necessary, and installing the connectivity component includes automatically without user intervention receiving, detecting, and installing the connectivity component to enable connectivity to the host system using the new hardware device.

52. (original) The method as in claim 32 wherein the connectivity component received includes an updated version of a connectivity component stored on the client device before the connectivity component is installed by the installation module.

53. (original) The method as in claim 52 wherein detecting whether installation of the connectivity component is needed includes comparing a version of the updated connectivity component received with a version of the connectivity component stored on the client device.

54. (original) The method as in claim 32 wherein:  
receiving the connectivity component includes receiving version information from a remote server associated with a connectivity component;  
detecting whether installation of the connectivity component is needed includes detecting whether installation of the connectivity component is needed on the client device, and  
determining a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received when installation of the connectivity component is needed; and  
installing the connectivity component includes installing the connectivity component stored on the client device when the version information associated with the stored connectivity component is determined to be correct when compared against the version information received from the remote server.

55. (original) The method as in claim 54 wherein:  
receiving the connectivity component includes receiving an updated connectivity component from the remote server when the version information associated with the connectivity component stored on the client device is determined not to be correct when compared against the version information received from the remote server; and  
installing the connectivity component includes installing the updated connectivity component received from the remote server.

56. (canceled).

57. (canceled).

58. (original) The method as in claim 32 wherein:

receiving the connectivity component includes using a host system to receive a request to send a connectivity component to a local client device; and

installing the connectivity component includes using the host system to send the connectivity component to the local client device for installation on the local client device in response to the request.

59. (original) The method as in claim 58 wherein detecting whether installation of the connectivity component is necessary includes using the host system to determine a version of the connectivity component to install on the local client device.

60. (previously presented) A computer program for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, the computer program being stored on a computer readable medium or a propagated signal and comprising:

a receiving code segment that causes the computer to receive a connectivity component that enables connectivity to a host system by at least one of several different hardware devices;

a detection code segment that causes the computer to detect whether installation of the connectivity component is needed to enable connectivity between the client device and the host system using a selected hardware device; and

an installation code segment that causes the computer to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device, wherein:

the detection code segment causes the computer to detect a new hardware device and, based on detecting the new hardware device, to determine whether a connectivity



component is stored locally that is needed to enable connectivity between the client device and the host system using the new hardware device;

the receiving code segment causes the computer to receive an updated connectivity component from a remote server when the detection code segment determines that the connectivity component that is needed to enable connectivity between the client device and the host system using the new hardware device is not stored locally; and

the installation code segment causes the computer to install the updated connectivity component received from the remote server.

61-66. (canceled).

67. (previously presented) The computer program of claim 60 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the new hardware device associated with the device driver.

68. (previously presented) The computer program of claim 60 wherein the connectivity component is capable of interfacing directly with the new hardware device to enable connectivity between the client device and the host system using the new hardware device.

69. (previously presented) The computer program of claim 60 wherein the connectivity component includes computer software that interfaces with a driver for the new hardware device that is used to connect to the host system.

70. (original) The computer program of claim 60 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

71. (canceled).

72. (canceled).

73. (original) The computer program of claim 60 wherein the installation code segment causes the computer to install a list of programs needed to install the connectivity component.

74. (original) The computer program of claim 73 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

75-78. (canceled).

79. (previously presented) The computer program of claim 60 wherein the receiving code segment, the detection code segment, and the installation code segment cause the computer to perform automatically without user intervention to receive, detect, and install the connectivity component to enable connectivity to the host system using the new hardware device.

80. (original) The computer program of claim 60 wherein the connectivity component received includes an updated version of a connectivity component stored on the client device before the connectivity component is installed by the installation code segment.

81. (original) The computer program of claim 80 wherein the detection code segment causes the computer to detect whether installation of the connectivity component is needed on the client device by comparing a version of the updated connectivity component received with a version of the connectivity component stored on the client device.

82. (original) The computer program of claim 60 wherein:  
the receiving code segment causes the computer to receive version information from a remote server associated with a connectivity component;

the detection code segment causes the computer to detect whether installation of the connectivity component is needed on the client device, and to determine a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received by the receiving code segment when installation of the connectivity component is needed; and

the installation code segment causes the computer to install the connectivity component stored on the client device when the detection code segment determines the version information associated with the stored connectivity component is correct when compared against the version information received from the remote server.

83. (original) The computer program of claim 82 wherein:

the receiving code segment causes the computer to receive an updated connectivity component from the remote server when the detection code segment determines that the version information associated with the connectivity component stored on the client device is not correct when compared against the version information received from the remote server; and

the installation code segment causes the computer to install the updated connectivity component received from the remote server.

84. (canceled).

85. (canceled).

86. (original) The computer program of claim 60 wherein:

the receiving code segment includes a host system receiving code segment that causes the computer to receive a request to send a connectivity component to a local client device; and

the installation code segment includes a host system installation code segment that causes the computer to send the connectivity component to the local client device for installation on the local client device in response to the request.

87. (original) The computer program of claim 86 wherein the host system detection code segment causes the computer to determine a version of the connectivity component to install on the local client device.

88. (previously presented) A system for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, comprising:

a receiving module that is structured and arranged to receive multiple connectivity components that enable connectivity to a host system by at least one of several different hardware devices, wherein the receiving module is structured and arranged to copy the connectivity components to the client device from a compact disk and store the connectivity components in a dormant state on the client device;

a detection module that is structured and arranged to detect whether installation of at least one of the connectivity components is needed to enable connectivity between the client device and the host system using a selected hardware device; and

an installation module that is structured and arranged to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device.

89. (previously presented) The system of claim 88 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the selected hardware device associated with the device driver.

90. (previously presented) The system of claim 88 wherein the connectivity component is capable of interfacing directly with the selected hardware device to enable connectivity between the client device and the host system using the selected hardware device.

91. (previously presented) The system of claim 88 wherein the connectivity component includes computer software that interfaces with a driver for the selected hardware device that is used to connect to the host system.

92. (previously presented) The system of claim 88 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

93. (previously presented) The system of claim 88 wherein:  
the detection module is structured and arranged to detect whether the installation of the connectivity component is needed to enable connectivity between the client device and the host system in response to an input received from a user of the client device requesting communications using the selected hardware device; and  
the installation module is structured and arranged to install the connectivity component when the connectivity component is needed based on the input from the user of the client device.

94. (previously presented) The system of claim 93 wherein the input from the user of the client device includes a request to change connectivity to the host system from a low-bandwidth connection type to a broadband connection type.

95. (previously presented) The system of claim 88 wherein the installation module installs a list of programs needed to install the connectivity component.

96. (previously presented) The system of claim 95 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

97. (previously presented) The system of claim 88 wherein the detection module further comprises an automatic hardware device detector that is structured and arranged to automatically

detect a hardware device and determine the connectivity component needed to enable connectivity to the host system associated with the hardware device detected.

98. (previously presented) The system of claim 88 wherein the receiving module, the detection module, and the installation module are structured and arranged to perform automatically without user intervention to receive, detect, and install the connectivity component to enable connectivity to the host system using the selected hardware device.

99. (previously presented) The system of claim 88 wherein the connectivity component received includes an updated version of a connectivity component stored on the client device before the connectivity component is installed by the installation module.

100. (previously presented) The system of claim 99 wherein the detection module is structured and arranged to detect whether installation of the connectivity component is needed on the client device by comparing a version of the updated connectivity component received with a version of the connectivity component stored on the client device.

101. (previously presented) The system of claim 88 wherein:  
the receiving module is structured and arranged to receive version information from a remote server associated with a connectivity component;  
the detection module is structured and arranged to detect whether installation of the connectivity component is needed on the client device, and to determine a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received by the receiving module when installation of the connectivity component is needed; and  
the installation module is structured and arranged to install the connectivity component stored on the client device when the detection module determines the version information associated with the stored connectivity component is correct when compared against the version information received from the remote server.

102. (previously presented) The system of claim 101 wherein:

the receiving module is structured and arranged to receive an updated connectivity component from the remote server when the detection module determines that the version information associated with the connectivity component stored on the client device is not correct when compared against the version information received from the remote server; and

the installation module is structured and arranged to install the updated connectivity component received from the remote server.

103. (previously presented) The system of claim 88 wherein:

the receiving module is structured and arranged to include a host system receiving module that is structured and arranged to receive a request to send a connectivity component to a local client device; and

the installation module is structured and arranged to include a host system installation module that is structured and arranged to send the connectivity component to the local client device for installation on the local client device in response to the request.

104. (previously presented) The system of claim 103 wherein the detection module is structured and arranged to include a host system detection module that is structured and arranged to determine a version of the connectivity component needed for installation on the local client device.

105. (previously presented) A method for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, the method comprising:

receiving multiple connectivity components that enable connectivity to a host system by at least one of several different hardware devices, wherein receiving the connectivity components includes copying the connectivity components to the client device from a compact disk and storing the connectivity components in a dormant state on the client device;

detecting whether installation of at least one of the connectivity components is needed to enable connectivity between the client device and the host system using a selected hardware device; and

installing the connectivity component when connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device.

106. (previously presented) The method as in claim 105 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the selected hardware device associated with the device driver.

107. (previously presented) The method as in claim 105 wherein the connectivity component is capable of interfacing directly with the selected hardware device to enable connectivity between the client device and the host system using the selected hardware device.

108. (previously presented) The method as in claim 105 wherein the connectivity component includes computer software that interfaces with a driver for the selected hardware device that is used to connect to the host system.

109. (previously presented) The method as in claim 105 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

110. (previously presented) The method as in claim 105 wherein:  
detecting whether installation of the connectivity component is necessary includes detecting whether the installation of the connectivity component is needed to enable connectivity between the client device and the host system in response to an input received from a user of the client device requesting communications using the selected hardware device; and  
installing the connectivity component includes installing the connectivity component when the connectivity component is needed based on the input from the user of the client device.



111. (previously presented) The method as in claim 110 wherein the input from the user of the client device includes a request to change connectivity to the host system from a low-bandwidth connection type to a broadband connection type.

112. (previously presented) The method as in claim 105 wherein installing the connectivity component includes installing a list of programs needed to install the connectivity component.

113. (previously presented) The method as in claim 112 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

114. (previously presented) The method as in claim 105 wherein detecting whether installation of the connectivity component is necessary further comprises automatically detecting a hardware device and determining the connectivity component needed to enable connectivity to the host system associated with the hardware device detected.

115. (previously presented) The method as in claim 105 wherein receiving the connectivity component, detecting whether installation of the connectivity component is necessary, and installing the connectivity component includes automatically without user intervention receiving, detecting, and installing the connectivity component to enable connectivity to the host system using the selected hardware device.

116. (previously presented) The method as in claim 105 wherein:  
receiving the connectivity component includes receiving version information from a remote server associated with a connectivity component;

detecting whether installation of the connectivity component is needed includes detecting whether installation of the connectivity component is needed on the client device, and determining a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received when installation of the connectivity component is needed; and

installing the connectivity component includes installing the connectivity component stored on the client device when the version information associated with the stored connectivity component is determined to be correct when compared against the version information received from the remote server.

117. (previously presented) The method as in claim 116 wherein:

receiving the connectivity component includes receiving an updated connectivity component from the remote server when the version information associated with the connectivity component stored on the client device is determined not to be correct when compared against the version information received from the remote server; and

installing the connectivity component includes installing the updated connectivity component received from the remote server.

118. (previously presented) The method as in claim 105 wherein:

receiving the connectivity component includes using a host system to receive a request to send a connectivity component to a local client device; and

installing the connectivity component includes using the host system to send the connectivity component to the local client device for installation on the local client device in response to the request.

119. (previously presented) The method as in claim 118 wherein detecting whether installation of the connectivity component is necessary includes using the host system to determine a version of the connectivity component to install on the local client device.

120. (previously presented) A computer program for installing computer software components on a client device for enabling connectivity to a host system by at least one of several different hardware devices, the computer program being stored on a computer readable medium or a propagated signal and comprising:

a receiving code segment that causes the computer to receive multiple connectivity components that enable connectivity to a host system by at least one of several different hardware devices, wherein the receiving code segment causes the computer to copy at least one of the connectivity components to the client device from a compact disk and store the connectivity component in a dormant state on the client device;

a detection code segment that causes the computer to detect whether installation of the connectivity component is needed to enable connectivity between the client device and the host system using a selected hardware device; and

an installation code segment that causes the computer to install the connectivity component when the connectivity component is needed to enable connectivity between the client device and the host system using the selected hardware device.

121. (previously presented) The computer program of claim 120 wherein the connectivity component is capable of interfacing a device driver to enable communications between computer software at the client device and the host system using the selected hardware device associated with the device driver.

122. (previously presented) The computer program of claim 120 wherein the connectivity component is capable of interfacing directly with the selected hardware device to enable connectivity between the client device and the host system using the selected hardware device.

123. (previously presented) The computer program of claim 120 wherein the connectivity component includes computer software that interfaces with a driver for the selected hardware device that is used to connect to the host system.

124. (previously presented) The computer program of claim 120 wherein the connectivity component includes a broadband connectivity component to enable connectivity to the host system using a broadband communication device.

125. (previously presented) The computer program of claim 120 wherein:  
the detection code segment causes the computer to detect whether the installation of the connectivity component is needed to enable connectivity between the client device and the host system in response to an input received from a user of the client device requesting communications using the selected hardware device; and  
the installation code segment causes the computer to install the connectivity component when the connectivity component is needed based on the input from the user of the client device.

126. (currently amended) The computer program of claim ~~[[120]]~~ 125 wherein the input from the user of the client device includes a request to change connectivity to the host system from a low-bandwidth connection type to a broadband connection type.

127. (previously presented) The computer program of claim 120 wherein the installation code segment causes the computer to install a list of programs needed to install the connectivity component.

128. (currently amended) The computer program of claim ~~[[120]]~~ 127 wherein the list of programs includes a sequential list of programs needed to install the connectivity component such that only one reboot of the client device is necessary to accomplish installation using the sequential list of programs.

129. (previously presented) The computer program of claim 120 wherein the detection code segment further comprises an automatic hardware device detector code segment that causes the computer to automatically detect a hardware device and determine the connectivity component needed to enable connectivity to the host system associated with the hardware device detected.

130. (previously presented) The computer program of claim 120 wherein the receiving code segment, the detection code segment, and the installation code segment cause the computer to perform automatically without user intervention to receive, detect, and install the connectivity component to enable connectivity to the host system using the selected hardware device.

131. (previously presented) The computer program of claim 120 wherein the connectivity component received includes an updated version of a connectivity component stored on the client device before the connectivity component is installed by the installation code segment.

132. (previously presented) The computer program of claim 131 wherein the detection code segment causes the computer to detect whether installation of the connectivity component is needed on the client device by comparing a version of the updated connectivity component received with a version of the connectivity component stored on the client device.

133. (previously presented) The computer program of claim 120 wherein:  
the receiving code segment causes the computer to receive version information from a remote server associated with a connectivity component;

the detection code segment causes the computer to detect whether installation of the connectivity component is needed on the client device, and to determine a version of the connectivity component to install by comparing the version information received from the remote server with version information associated with the connectivity component already received by the receiving code segment when installation of the connectivity component is needed; and

the installation code segment causes the computer to install the connectivity component stored on the client device when the detection code segment determines the version information associated with the stored connectivity component is correct when compared against the version information received from the remote server.

134. (previously presented) The computer program of claim 133 wherein:

the receiving code segment causes the computer to receive an updated connectivity component from the remote server when the detection code segment determines that the version information associated with the connectivity component stored on the client device is not correct when compared against the version information received from the remote server; and

the installation code segment causes the computer to install the updated connectivity component received from the remote server.

135. (previously presented) The computer program of claim 120 wherein:

the receiving code segment includes a host system receiving code segment that causes the computer to receive a request to send a connectivity component to a local client device; and

the installation code segment includes a host system installation code segment that causes the computer to send the connectivity component to the local client device for installation on the local client device in response to the request.

136. (previously presented) The computer program of claim 135 wherein the host system detection code segment causes the computer to determine a version of the connectivity component to install on the local client device.

137. (previously presented) The system of claim 1 wherein the detection module is structured and arranged to detect a prior receipt of the connectivity component that is needed to enable connectivity between the client device and the host system using the new hardware device.

138. (previously presented) The method as in claim 32 wherein detecting whether installation of the connectivity component is necessary includes detecting a prior receipt of the connectivity component that is needed to enable connectivity between the client device and the host system using the new hardware device.

139. (previously presented) The computer program of claim 60 wherein the detection code segment causes the computer to detect a prior receipt of the connectivity component that is needed to enable connectivity between the client device and the host system using the new hardware device.

Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 40 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

**IX. Evidence Appendix**

None.



Applicant : Tuan Huu Pham et al.  
Serial No. : 09/855,683  
Filed : May 16, 2001  
Page : 41 of 41

Attorney's Docket No.: 06975-  
136001 / Communications 38

**X. Related Proceedings Appendix**

None.